

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Cancelled)

2. (Currently Amended) A ~~computer readable medium~~ method as recited in claim 31, ~~having further comprising computer executable instructions for performing the step of~~ reviewing configuration information to identify the ~~predefined attribute~~ relationship associated with the view name in the location path expression.

3. (Currently Amended) A ~~computer readable medium~~ method as recited in claim 2, wherein ~~the step of~~ reviewing configuration information further identifies a root level starting point associated with the view name.

4. (Currently Amended) A ~~computer readable medium~~ method as recited in claim 2, wherein ~~the step of~~ reviewing the configuration determines whether the client has permission to access the database based on the ~~predefined attribute~~ relationship.

5. (Cancelled)

6. (Cancelled)

7. (Currently Amended) A ~~computer readable medium~~ method as recited in claim 31, wherein ~~the step of receiving receives the database access client request in a message is received according to the Simple Object Access Protocol (SOAP).~~

8. (Currently Amended) A ~~computer readable medium~~ method as recited in claim 31, wherein one of the at least one path elements of the location path expression is a wildcard element.

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

9. (Currently Amended) A ~~computer-readable medium~~ method as recited in claim 31, wherein one of the at least one path elements of the location path expression indicates a search in a reversed direction of the predefined relationship.

10. (Cancelled)

11. (Cancelled)

12. (Currently Amended) A ~~computer-readable medium~~ method as recited in claim 31, wherein the database is a directory service database.

13. (Cancelled).

14. (Currently Amended) A ~~computer-readable medium~~ method as recited in claim 138, having further comprising computer executable instructions for performing the step of obtaining configuration information from the server defining the relationships among linking attributes of the objects in the database and associated view names thereof.

15. (Currently Amended) A ~~computer-readable medium~~ method as recited in claim 14, wherein the step of sending the request sends the request in a message to the server according to the Simple Object Access Protocol (SOAP).

16. (Currently Amended) A ~~computer-readable medium~~ method as recited in claim 138, wherein one of the at least one path elements of the location path expression is a wildcard element.

17. (Cancelled).

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

18. (Currently Amended) A ~~computer-readable medium~~ computer program product as recited in claim 437, having further computer-executable instructions for performing the step of reviewing configuration information to identify the predefined attribute relationship associated with the view name in the location path expression.

19. (Currently Amended) A ~~computer-readable medium~~ computer program product as in claim 18, wherein ~~the step of~~ reviewing configuration information further identifies a root level starting point associated with the view name.

20. (Currently Amended) A ~~computer-readable medium~~ computer program product as recited in claim 18, wherein ~~the step of~~ reviewing the configuration determines whether the client has permission to access the database based on the predefined relationship.

21. (Currently Amended) A ~~computer-readable medium~~ computer program product as recited in claim 437, wherein the database is a directory service database.

22-28. (Cancelled).

29. (Currently Amended) A ~~database system~~method as recited in claim 2638, wherein the server is a database server of the database.

30. (Currently Amended) A ~~database system~~method as recited in claim 26, wherein the database is a directory service database.

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

31. (New) A method for accessing objects in a database, comprising:
- storing objects in a database, wherein the objects each comprise corresponding attributes;
- defining relationships linking attributes of the objects, wherein defining the relationships includes creating pointers linking each object having a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein defining attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;
- receiving a client request for accessing a requested object in the database, wherein the request is entered in the format of a location path expression having the following format:
- a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and
- at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;
- processing the client request comprising the location path expression to locate the requested object; and
- returning the requested object and any other data specified in the location path expression to a client.

32. (New) A method as recited in claim 31, wherein the database is a database of a Web service, and wherein the location path expression is translated into a plurality of LDAP queries that are processed by the Web service to satisfy the client request and that are iteratively processed until the client request is satisfied.

33. (New) A method as recited in claim 31, wherein the location path expression includes a plurality of objects related by the defined attribute relationship specified by the view name, and wherein each of the objects are separated by a forward slash.

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

34. (New) A method as recited in claim 31, where in at least one of the defined attribute relationships includes a relationship between objects of different types that are linked by an attribute relationship.

35. (New) A method as recited in claim 31, where in the location path expression is written as an abbreviated XPath expression.

36. (New) A method as recited in claim 31, wherein the method further includes: providing an application programming interface (API) from which applications on the client issue function calls to form the data path expression and to send the data path expression over a transport protocol to a Web service for directory access to the database.

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

37. (New) A computer program product for implementing a method for accessing objects in a database, the computer program product comprising:

one or more physical computer-readable media having stored thereon computer-executable instructions that, when executed by a processor, cause a computing system to perform the following:

store objects in a database, wherein the objects each comprise corresponding attributes;

define relationships linking attributes of the objects, wherein defining the relationships includes creating pointers linking each object having a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein defining attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

receive a client request for accessing a requested object in the database, wherein the request is entered in the format of a location path expression having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

process the client request comprising the location path expression to locate the requested object; and

return the requested object and any other data specified in the location path expression to a client.

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

38. (New) A method for receiving objects requested from a database, the method comprising:

connecting with a server providing access to objects stored in a database, wherein the objects each comprise corresponding attributes, and wherein relationships linking attributes of the objects are defined by creating pointers linking each object having a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein the defined attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

forming a request for access to a requested object in the database, wherein the request is entered in the format of a location path expression having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

sending the request comprising the location expression to cause the server to locate the requested object; and

receiving the requested object and any other data specified in the location path expression.

Application No. 10/620,095
Amendment "A" dated May 4, 2006
Reply to Office Action mailed January 4, 2006

39. (New) A computer program product for implementing a method for accessing objects in a database, the computer program product comprising:

one or more physical computer-readable media having stored thereon computer-executable instructions that, when executed by a processor, cause a computing system to perform the following:

connect with a server providing access to objects stored in a database, wherein the objects each comprise corresponding attributes, and wherein relationships linking attributes of the objects are defined by creating pointers linking each object having a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein the defined attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

form a request for access to a requested object in the database, wherein the request is entered in the format of a location path expression having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

send the request comprising the location expression to cause the server to locate the requested object; and

receive the requested object and any other data specified in the location path expression.